

Decision problems for word-hyperbolic semigroups

Alan J. Cain

Centro de Matemática, Universidade do Porto,
Rua do Campo Alegre 687, 4169-007 Porto, Portugal

Email: ajcain@fc.up.pt

Web page: www.fc.up.pt/pessoas/ajcain/

ABSTRACT

This paper studies decision problems for semigroups that are word-hyperbolic in the sense of Duncan & Gilman. A fundamental investigation reveals that the natural definition of a ‘word-hyperbolic structure’ has to be strengthened slightly in order to define a unique semigroup up to isomorphism. It is proved that every word-hyperbolic semigroup has word problem solvable in polynomial time (improving on the previous exponential-time algorithm). Algorithms are presented for deciding whether a word-hyperbolic semigroup is a monoid, a group, a completely simple semigroup, a Clifford semigroup, or a free semigroup.

1 INTRODUCTION

The concept of word-hyperbolicity in groups, which has grown into one of the most fruitful areas of group theory since the publication of Gromov’s seminal paper [Gro87], admits a natural extension to monoids via using Gilman’s characterization of word-hyperbolic groups using context-free languages [Gil02], which generalizes directly to semigroups and monoids [DGo4]. Informally, a word-hyperbolic structure for a semigroup consists of a regular language of representatives (not necessarily unique) for the elements of the semigroup, and a context-free language describing the multiplication table of the semigroup in terms of those representatives.

This generalizaion has led to a substantial amount of research on word-hyperbolic semigroups; see, for example, [CM12, FK04, HKOT02, HT03]. Some of this work has shown that word-hyperbolic semigroups do not possess such pleasant properties as word-hyperbolic groups: they may not be finitely presented; they are not in general automatic or even asynchronously automatic [HKOT02, Example 7.7 et seq.].

Acknowledgements: During the research that led to the this paper, the author was supported by the European Regional Development Fund through the programme COMPETE and by the Portuguese Government through the FCT (Fundação para a Ciência e a Tecnologia) under the project PEst-C/MAT/UI0144/2011 and through an FCT Ciência 2008 fellowship.

The computational aspect of word-hyperbolic semigroups has so far received limited attention. The only established result seems to be the solvability of the word problem [HKOT02, Theorem 3.8]. In contrast, automatic semigroups, which generalize automatic groups [ECH⁺92] and whose study was inaugurated by Campbell et al. [CRR01], have been studied from a computational perspective, with both decidability and undecidability results emerging [Caio6, KO06, Otto7].

This paper is devoted to some important decision problems for word-hyperbolic semigroups. Word-hyperbolic structures are not necessarily ‘stronger’ or ‘weaker’ computationally than automatic structures. As noted above, word-hyperbolicity does not imply automaticity for semigroups, so one cannot appeal to known results for automatic semigroups. A word-hyperbolic structure encodes the whole multiplication table for the semigroup, not just right-multiplication by generators (as is the case for automatic structures). On the other hand, context-free languages are less computationally pleasant than regular languages. For instance, an intersection of two context-free languages is not in general context-free, and indeed the emptiness of such an intersection cannot be decided algorithmically. Thus, in constructing algorithms for word-hyperbolic semigroups, it is often necessary to proceed via an indirect route, or use some unusual ‘trick’.

The most important result is a polynomial-time algorithm for solving the word problem (Section 4). As remarked above, the word problem was already known to be solvable, but the previously-known algorithm required time exponential in the lengths of the input words [HKOT02, Theorem 3.8].

Some basic properties are then shown to be decidable (Section 5): being a monoid, Green’s relations \mathcal{L} , \mathcal{R} , and \mathcal{H} , being a group, and commutativity. These results are not particularly difficult, but are worth noting.

The main body of the paper shows the decidability of more complicated algebraic properties: being completely simple (Section 6), being a Clifford semigroup (Section 7), and being a free semigroup (Section 8).

Before embarking on the discussion of decision problems, it is necessary to make a fundamental study of the notion of word-hyperbolicity, because the natural notion of a word-hyperbolic structure, or more precisely an ‘interpretation’ of a word-hyperbolic structure, does not determine a unique semigroup up to isomorphism. A slightly strengthened definition is needed, and this is the purpose of the preliminary Section 2.

The paper ends with a list of some open problem (Section 9).

2 THE LIMITS OF INTERPRETATION

Before developing any algorithms for word-hyperbolic semigroups, we must clarify the relationship between a word-hyperbolic structure (that is, an abstract collection of certain languages) and a semigroup it describes. A similar study grounds the study of decision problems for automatic semigroups by Kambites & Otto [KO06], and our strategy and choice of terminology closely follows theirs.

Let A be an alphabet. For any word $w \in A^*$, the reverse of w is denoted w^{rev} . We extend this notation to languages: for any language $L \subseteq A^*$, let $L^{\text{rev}} = \{w^{\text{rev}} : w \in L\}$.

DEFINITION 2.1. A *pre-word-hyperbolic structure* Σ consists of:

- a finite alphabet $A(\Sigma)$;
- a regular language $L(\Sigma)$ over $A(\Sigma)$;
- a context-free language $M(\Sigma)$ over $A(\Sigma) \cup \{\#_1, \#_2\}$, where $\#_1$ and $\#_2$ are new symbols not in $A(\Sigma)$, such that $M(\Sigma) \subseteq L(\Sigma)\#_1 L(\Sigma)\#_2 L(\Sigma)^{\text{rev}}$.

When Σ is clear from the context, we may write A , L , and M instead of $A(\Sigma)$, $L(\Sigma)$, and $M(\Sigma)$, respectively.

‘Pre-word-hyperbolic structures’ consist only of languages fulfilling certain basic properties: there is no mention of being a structure ‘for a semigroup’ in the definition. Now, following Kambites & Otto for automatic semigroups [KOO6, § 2.2], let us make a first attempt to turn the abstract pre-word-hyperbolic structure into something that describes a semigroup. As we shall see, this definition is flawed, but it is instructive to see its consequences, since these illustrate why the improved [Definition 2.7](#) is actually the correct one.

DEFINITION 2.2 ((First attempt)). An *interpretation* of a pre-word-hyperbolic structure Σ with respect to a semigroup S is a map $\phi : A^+ \rightarrow S$ such that $L\phi = S$ and

$$M(\Sigma) = \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L, (u\phi)(v\phi) = w\phi\}.$$

When there is no risk of confusion, denote $u\phi$ by \bar{u} for any $u \in A^+$, and $X\phi$ by \bar{X} for any $X \subseteq A^+$.

If a pre-word-hyperbolic structure Σ admits an interpretation with respect to a semigroup S , then Σ is a *word-hyperbolic structure* for S .

Suppose Σ is a word-hyperbolic structure for S , as per [Definition 2.2](#). Then words in A^+ represent elements of S , the regular language L contains at least one representative for every element of S , and the context-free language M encodes the multiplication table for S in terms of representatives in L . However, there is a problem. In contrast to the situation for automatic semigroups [KOO6, Proposition 2.3], a word-hyperbolic structure (using [Definition 2.2](#)) does not uniquely determine a semigroup: the same pre-word-hyperbolic structure can admit interpretations with respect to non-isomorphic semigroups, as the following example shows:

EXAMPLE 2.3. Let Σ be a pre-word-hyperbolic structure with $A = \{a, b, c\}$, $L = A$, and $M = \{u\#_1 v\#_2 a^{\text{rev}} : u, v \in L\}$. (Of course, $a^{\text{rev}} = a$ since a is a single letter.)

Let S be the two-element null semigroup $\{0, x\}$, where all products are equal to 0. Let T be the three-element null semigroup $\{0, x, y\}$, again with all products equal to 0.

Define mappings $\phi : A \rightarrow S$ and $\psi : A \rightarrow T$ by

$$\begin{array}{lll} a\phi = 0, & b\phi = x, & c\phi = x, \\ a\psi = 0, & b\psi = x, & c\psi = y. \end{array}$$

Then $L\phi = S$ and $L\psi = T$. Furthermore,

$$\begin{aligned} M &= \{u\#_1 v\#_2 a^{\text{rev}} : u, v \in L\} \\ &= \{u\#_1 v\#_2 a^{\text{rev}} : u, v \in L, (u\phi)(v\phi) = a\phi\} \\ &= \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L, (u\phi)(v\phi) = w\phi\}, \end{aligned}$$

since all products in S are equal to 0 and a is the unique word in L mapped to 0 by ϕ . Similarly,

$$M = \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L, (u\psi)(v\psi) = w\psi\}$$

since all products in T are equal to 0 and a is the unique word in L mapped to 0 by ψ .

Thus ϕ and ψ are interpretations of Σ with respect to the non-isomorphic semigroups S and T respectively.

Hence, it seems not to make sense to consider decision problems for general word-hyperbolic semigroups, at least with the current definitions. It would be illogical to ask for an algorithm that takes as input a word-hyperbolic structure and determined some property of ‘the’ semigroup it describes, since there is no such unique semigroup. The fundamental problem [Example 2.3](#) elucidates is that the word-hyperbolic structure Σ does not determine whether the two symbols b and c represent the same element or different elements. However, this problem only arises for two symbols in A ; that is, for two words over A of length 1, in the sense that if we insist that an interpretation should consist of a map that sends distinct letters to distinct elements of the semigroup (that is, the map is an injection when restricted to A), then a word-hyperbolic structure does describe a unique semigroup up to isomorphism. In order to prove this result ([Proposition 2.5](#)), we need the following lemma:

LEMMA 2.4. *Let Σ be a word-hyperbolic structure. Then there is a relation $E(\Sigma) \subseteq L(\Sigma) \times L(\Sigma)$, dependent only on Σ , such that the following are equivalent for any words $w, x \in L(\Sigma)$:*

1. $(w, x) \in E(\Sigma)$;
2. $w\phi = x\phi$ for some interpretation $\phi : A(\Sigma)^+ \rightarrow S$ with $\phi|_A$ being an injection;
3. $w\phi = x\phi$ for any interpretation $\phi : A(\Sigma)^+ \rightarrow S$ with $\phi|_A$ being an injection.

Proof of 2.4. Define

$$E' = \{(w, x) : w \in L(\Sigma), x \in L(\Sigma), |w| \geq |x|, |w| \geq 2, \\ (\exists u, v \in L(\Sigma))(u\#_1 v\#_2 w^{\text{rev}} \in M(\Sigma) \wedge u\#_1 v\#_2 x^{\text{rev}} \in M(\Sigma))\}$$

and let

$$E(\Sigma) = \{(a, a) : a \in A(\Sigma) \cap L(\Sigma)\} \cup E' \cup (E')^{-1}. \quad (2.1)$$

The aim is now to show that $E(\Sigma)$ has the required properties. Let $w, x \in L(\Sigma)$.

First suppose that (1) holds; that is, that $(w, x) \in E(\Sigma)$. Let ϕ be any interpretation of Σ . Either $w, x \in A(\Sigma) \cap L(\Sigma)$, in which case $w = x$ by (2.1) and so $w\phi = x\phi$, or $(w, x) \in E' \cup (E')^{-1}$. Assume $(w, x) \in E'$; the other case is symmetrical. Then there exist $u, v \in L(\Sigma)$ such that $u\#_1 v\#_2 w^{\text{rev}} \in M(\Sigma)$ and $u\#_1 v\#_2 x^{\text{rev}} \in M(\Sigma)$. Hence $w\phi = (u\phi)(v\phi) = x\phi$ since ϕ is an interpretation of Σ . Hence (1) implies (3).

It is clear that (3) implies (2). Now suppose that (2) holds; that is, that $w\phi = x\phi$ for some interpretation $\phi : A(\Sigma)^+ \rightarrow S$ with $\phi|_A$ being an injection. If $|w| = |x| = 1$, then $w, x \in A$ and so $w = x$ since $\phi|_A$ is injective, and so $(w, x) \in E(\Sigma)$. Now suppose that at least one of $|w|$ and $|x|$ is greater than 1. Assume $|w| \geq |x|$; the other case is similar. Since w has at least two letters,

the element $w\phi$ is decomposable in S . So there are words $u, v \in L$ with $(u\phi)(v\phi) = w\phi$. Since $w\phi = x\phi$, it also follows that $(u\phi)(v\phi) = x\phi$. Thus the words $u\#_1 v\#_2 w^{\text{rev}}$ and $u\#_1 v\#_1 x^{\text{rev}}$ both lie in M since ϕ is an interpretation of Σ . Hence $(w, x) \in E' \subseteq E(\Sigma)$. Hence (2) implies (1). [2.4]

PROPOSITION 2.5. *Let Σ be a word-hyperbolic structure admitting interpretations $\phi : A^+ \rightarrow S$ and $\psi : A^+ \rightarrow T$, with both $\phi|_A$ and $\psi|_A$ being injections. Then there is an isomorphism τ from S to T such that $\phi|_L \tau = \psi|_L$.*

Proof of 2.5. Define maps $\tau : S \rightarrow T$ and $\tau' : T \rightarrow S$ as follows. For any $s \in S$ let $s\tau$ be $w\psi$, where $w \in L$ is some word with $w\phi = s$, and for any $t \in T$, let $t\tau'$ be $w'\phi$, where $w' \in L$ is some word with $w'\psi = t$. (The words w and w' are guaranteed to exist since ϕ and ψ are surjections.) The maps τ and τ' are well-defined as a consequence of [Lemma 2.4](#).

To show that τ is a homomorphism, proceed as follows. Let $r, s \in S$ and choose $u, v, w \in L$ with $u\phi = r$, $v\phi = s$, and $w\phi = rs$. Then $r\tau = u\psi$, $s\tau = v\psi$, and $(rs)\tau = w\psi$, by the definition of τ . Now, $u\#_1 v\#_2 w^{\text{rev}} \in M$ (since ϕ is an interpretation of Σ) and so $(u\psi)(v\psi) = w\psi$ (since ψ is an interpretation of Σ). Thus

$$(r\tau)(s\tau) = (u\psi)(v\psi) = (w\psi) = (rs)\tau$$

and so τ is a homomorphism.

Symmetric reasoning shows that $\tau' : T \rightarrow S$ is a homomorphism. The maps τ and τ' are mutually inverse, since if $w \in L$ is such that $w\phi = s$ and $w\psi = t$, then $s\tau = t$ and $t\tau' = s$. Thus $\tau : S \rightarrow T$ is an isomorphism. By the definition of τ using elements of L , it follows that $\phi|_L \tau = \psi|_L$. [2.5]

The extra condition used in [Proposition 2.5](#), where the interpretation restricted to the alphabet A is an injection, does not restrict the class of word-hyperbolic semigroups:

PROPOSITION 2.6. *Let Σ be a word-hyperbolic structure and $\phi : A(\Sigma)^+ \rightarrow S$ an interpretation for Σ with respect to a semigroup S . Then there is a word-hyperbolic structure Π , effectively computable from Σ and $\phi|_{A(\Sigma)}$, with $A(\Pi) \subseteq A(\Sigma)$, admitting an interpretation $\psi : A(\Pi)^+ \rightarrow S$ with $\psi|_{A(\Pi)}$ being an injection.*

Proof of 2.6. Initially, let $\Pi = \Sigma$. We will modify Π until it has the desired property.

Suppose $\phi|_{A(\Pi)}$ is not injective. Pick $a, b \in A(\Pi)$ with $a\phi = b\phi$. Replace every instance of b by a in words in $L(\Pi)$. (This corresponds to replacing b by a whenever it appears as a label on an edge in a finite automaton recognizing $L(\Pi)$.) Replace every instance of b by a in words in $M(\Pi)$. (This corresponds to replacing b by a whenever it appears as non-terminal in a context-free grammar defining $L(\Pi)$.) Finally, delete b from $A(\Pi)$. Since $a\phi = b\phi$, it follows that Π is a word-hyperbolic structure admitting an interpretation $\phi|_{A(\Pi)^+} : A(\Pi)^+ \rightarrow S$ with respect to S .

Since $A(\Pi)$ is finite, we can iterate this process until $\phi|_{A(\Pi)}$ becomes injective. Finally, define $\psi = \phi|_{A(\Pi)^+}$. [2.6]

In light of [Proposition 2.6](#), we modify our definition of interpretation, to insist that each symbol represent a different element of the semigroup:

DEFINITION 2.7 ((Improved version)). An *interpretation* of a pre-word-hyperbolic structure Σ with respect to a semigroup S is a map $\phi : A(\Sigma)^+ \rightarrow S$, with $\phi|_A$ being injective, such that $(L(\Sigma))\phi = S$ and

$$M(\Sigma) = \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L(\Sigma), (u\phi)(v\phi) = w\phi\}.$$

Again, when there is no risk of confusion, denote $u\phi$ by \bar{u} for any $u \in A^+$, and $X\phi$ by \bar{X} for any $X \subseteq A^+$.

Therefore, a word-hyperbolic structure is henceforth a pre-word-hyperbolic structure that admits an interpretation in the sense of Definition 2.7. With this new definition, Proposition 2.5 shows that each word-hyperbolic structure describes a uniquely determined semigroup, and therefore one can sensibly attempt to solve questions about the semigroup using the word-hyperbolic structure.

However, although a word-hyperbolic structure determines a unique semigroup, it does not determine a unique interpretation, even up to automorphic permutation. This parallels the situation for automatic semigroups [Koo6, § 2.2], but is also true in a rather vacuous sense for word-hyperbolic semigroups, for the alphabet $A(\Sigma)$ for a word-hyperbolic structure Σ for a semigroup S may include a symbol c that does not appear in any word in either $L(\Sigma)$ or $M(\Sigma)$. (In this situation, c must represent a redundant generator for S .)

For example, let $A(\Sigma) = \{a, b, c\}$, $L(\Sigma) = \{a, b\}^+$, and $M(\Sigma) = \{u\#_1 v\#_1 (uv)^{\text{rev}} : u, v \in \{a, b\}^+\}$. Then Σ is a word-hyperbolic structure for the free semigroup F with basis $\{x, y\}$: let $\phi : A(\Sigma)^+ \rightarrow F$ be such that $a\phi = x$ and $b\phi = y$; regardless of how $c\phi$ is defined, ϕ is an interpretation of Σ with respect to F .

Less trivial is the following example:

EXAMPLE 2.8. Let $S = (\{1, 2\} \times \{1, 2, 3\}) \cup \{0_S, 1_S\}$ and define multiplication on S by

$$(i, \lambda)(j, \mu) = \begin{cases} 0_S & \text{if } \lambda = j = 1, \\ (i, \mu) & \text{otherwise;} \end{cases}$$

$$1_S x = x 1_S = x \quad \text{for all } x \in S;$$

$$0_S x = x 0_S = 0_S \quad \text{for all } x \in S.$$

Then S is a monoid. (In fact, S is a monoid formed by adjoining an identity to a 0-Rees matrix semigroup over the trivial group.)

Let $A(\Sigma) = \{a, b, c, d, e, i, z\}$. Let $L(\Sigma) = \{a, b, c, d, bed, deb, i, z\}$. Define

$$\begin{aligned} \phi_1 : A(\Sigma) \rightarrow S \quad & a\phi_1 = (1, 1), \quad b\phi_1 = (1, 2), \quad c\phi_1 = (1, 3), \\ & d\phi_1 = (3, 1), \quad e\phi_1 = (2, 2), \quad i\phi_1 = 1_S, \quad z\phi_1 = 0_S; \\ \phi_2 : A(\Sigma) \rightarrow S \quad & a\phi_2 = (1, 1), \quad b\phi_2 = (1, 2), \quad c\phi_2 = (1, 3), \\ & d\phi_2 = (3, 1), \quad e\phi_2 = (1, 3), \quad i\phi_2 = 1_S, \quad z\phi_2 = 0_S. \end{aligned}$$

Define

$$M(\Sigma) = \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L(\Sigma), (u\phi_1)(v\phi_1) = w\phi_1\}.$$

Since $L(\Sigma)$ is finite, $M(\Sigma)$ is also finite and thus context-free. So Σ is a word-hyperbolic structure for S and ϕ_1 is an interpretation for Σ with respect to S . However,

$$(bed)\phi_1 = (1, 2)(2, 2)(2, 3) = (1, 3) = (1, 2)(1, 3)(2, 3) = (bed)\phi_2,$$

and similarly $(\text{deb})\phi_1 = (\text{deb})\phi_2$. Hence $\phi_1|_L = \phi_2|_L$. Therefore

$$M(\Sigma) = \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L(\Sigma), (u\phi_2)(v\phi_2) = w\phi_2\},$$

and so ϕ_2 is also an interpretation of Σ with respect to S . Moreover, there is no automorphism ρ of S such that $\phi_1\rho = \phi_2$. To see this, notice that such a ρ would have to map $e\phi_1 = (2, 2)$ to $e\phi_2 = (1, 3)$. The map ρ would also preserve \mathcal{R} -classes. But the \mathcal{R} -class of $(1, 3)$ contains the element $(1, 1)$, which is not idempotent (since $(1, 1)(1, 1) = 0$), whereas every element of the \mathcal{R} -class of $(2, 2)$ is idempotent. So no such map ρ can exist. So the two interpretations are not even equivalent up to automorphic permutation of S .

The crucial point in [Example 2.8](#) is that [Proposition 2.5](#) only guarantees that the *restriction* of two interpretations to L are equivalent up to automorphic permutation. It says nothing about the interpretation maps on the whole of A^+ .

In order to compute with the semigroup described by a word-hyperbolic structure, interpretations must be coded in a finite way.

DEFINITION 2.9. An *assignment of generators* for a word-hyperbolic structure Σ is a map $\alpha : A(\Sigma) \rightarrow L(\Sigma)$ with the property that there is some interpretation $\phi : A(\Sigma)^+ \rightarrow S$ such that $\alpha\alpha\phi = \alpha\phi$ for all $a \in A$; such an interpretation is said to be *consistent* with α . Two assignments of generators α and β for Σ are *equivalent* if $(\alpha\alpha, \alpha\beta) \in E(\Sigma)$ for all $a \in A(\Sigma)$.

PROPOSITION 2.10. An assignment of generators for a word-hyperbolic structure is consistent with a unique interpretation (up to automorphic permutation of the semigroup described). Equivalent assignments of generators are consistent with the same interpretation.

Conversely, every interpretation is consistent with a unique (up to equivalence) assignment of generators.

Proof of 2.10. Let Σ be a word-hyperbolic structure and $\alpha : A \rightarrow L$ an assignment of generators. Then there is an interpretation $\phi : A^+ \rightarrow S$ of Σ that is consistent with α ; that is, $\alpha\alpha\phi = \alpha\phi$ for all $a \in A$.

Let $\psi : A^+ \rightarrow S$ be another interpretation of Σ that is consistent with α ; the aim is to show that ϕ and ψ differ only by an automorphic permutation of S . First, $\alpha\alpha\psi = \alpha\psi$ for all $a \in A$, since ψ is consistent with α . By [Proposition 2.5](#), there is an automorphism τ of S such that $\phi|_L\tau = \psi|_L$, and so $\alpha\phi\tau = \alpha\alpha\phi\tau = \alpha\alpha\psi = \alpha\psi$ for all $a \in A$. Hence ϕ and ψ differ only by the automorphism τ .

Now let $\beta : A \rightarrow L$ be an assignment of generators equivalent to α ; the aim is to show that β is also consistent with ϕ . Now, $(\alpha\alpha, \alpha\beta) \in E(\Sigma)$ for all $a \in A$ since α and β are equivalent. Thus $\alpha\phi = \alpha\alpha\phi = \alpha\beta\phi$ by [Lemma 2.4](#), and hence β is also consistent with the interpretation ϕ .

Finally, suppose $\gamma : A \rightarrow L$ is an assignment of generators consistent with ϕ ; the aim is to show α and β are equivalent. Now, $\alpha\alpha\phi = \alpha\phi = \alpha\gamma\phi$ for all $a \in A$ since α and γ are consistent with ϕ . Hence $(\alpha\alpha, \alpha\gamma) \in E(\Sigma)$ for all $a \in A$ by [Lemma 2.4](#). 2.10

DEFINITION 2.11. A word-hyperbolic structure Σ is said to be an *interpreted* word-hyperbolic structure if it is equipped with an assignment of generators $\alpha(\Sigma)$.

PROPOSITION 2.12. *Let Σ be an interpreted word-hyperbolic structure for a semigroup S . Then there is another interpreted word-hyperbolic structure Σ' for S , effectively computable from Σ , such that $A(\Sigma') \subseteq L(\Sigma')$ and $\alpha(\Sigma')$ is the embedding map from $A(\Sigma')$ to $L(\Sigma')$.*

Proof of 2.12. Let $A(\Sigma')$ be $A(\Sigma)$ and $L(\Sigma') = L(\Sigma) \cup A(\Sigma)$. For brevity, let $\alpha = \alpha(\Sigma)$. For each $a, b, c \in A(\Sigma')$, define the languages:

$$\begin{aligned} M_a^{(1)} &= \{a\#_1 v\#_2 w^{\text{rev}} : (a\alpha)\#_1 v\#_2 w^{\text{rev}} \in M(\Sigma)\}, \\ M_a^{(2)} &= \{u\#_1 a\#_2 w^{\text{rev}} : u\#_1 (a\alpha)\#_2 w^{\text{rev}} \in M(\Sigma)\}, \\ M_a^{(3)} &= \{u\#_1 v\#_2 a^{\text{rev}} : u\#_1 v\#_2 (a\alpha)^{\text{rev}} \in M(\Sigma)\}, \\ M_{a,b}^{(4)} &= \{a\#_1 b\#_2 w^{\text{rev}} : (a\alpha)\#_1 (b\alpha)\#_2 w^{\text{rev}} \in M(\Sigma)\}, \\ M_{a,b}^{(5)} &= \{u\#_1 a\#_2 b^{\text{rev}} : u\#_1 (a\alpha)\#_2 (b\alpha)^{\text{rev}} \in M(\Sigma)\}, \\ M_{a,b}^{(6)} &= \{a\#_1 v\#_2 b^{\text{rev}} : (a\alpha)\#_1 v\#_2 (b\alpha)^{\text{rev}} \in M(\Sigma)\}, \\ M_{a,b,c}^{(7)} &= \{a\#_1 b\#_2 c^{\text{rev}} : (a\alpha)\#_1 (b\alpha)\#_2 (c\alpha)^{\text{rev}} \in M(\Sigma)\}. \end{aligned}$$

Each of these languages is context-free because each is the intersection of the context-free language $M(\Sigma)$ with a regular language. [Notice that $M_{a,b,c}^{(7)}$ is either empty or a singleton language.]

Now let

$$\begin{aligned} M(\Sigma') &= M(\Sigma) \cup \bigcup_{a \in A(\Sigma)} (M_a^{(1)} \cup M_b^{(1)} \cup M_c^{(1)}) \\ &\quad \cup \bigcup_{a,b \in A(\Sigma)} (M_{a,b}^{(1)} \cup M_{a,b}^{(1)}) \\ &\quad \cup \bigcup_{a,b,c \in A(\Sigma)} M_{a,b,c}^{(1)}; \end{aligned}$$

notice that $M(\Sigma')$ is also context-free.

Let $\phi : A(\Sigma) \rightarrow S$ be an interpretation of Σ . Then, recalling that $A(\Sigma) = A(\Sigma')$,

$$M(\Sigma') = \{u\#_1 v\#_2 w^{\text{rev}} : u, v, w \in L(\Sigma'), (u\phi)(v\phi) = w\phi\},$$

because u, v , and w range over $L(\Sigma') = L(\Sigma) \cup A(\Sigma)$, and the eight cases that arise depending on whether each word lies in $L(\Sigma)$ or $A(\Sigma)$ correspond to the eight sets $M(\Sigma)$, $M_a^{(1)}$, $M_a^{(2)}$, $M_a^{(3)}$, $M_{a,b}^{(4)}$, $M_{a,b}^{(5)}$, $M_{a,b}^{(6)}$, and $M_{a,b,c}^{(7)}$. [Notice that these sets are not necessarily disjoint, since it is possible that $a\alpha = a$ for some $a \in A$.]

Finally, define $\alpha(\Sigma')$ to be the embedding map from $A(\Sigma')$ to $L(\Sigma')$. This map is an assignment of generators since trivially $((a)\alpha(\Sigma'))\phi = a\phi$ for any interpretation ϕ of Σ' . [2.12]

In light of [Proposition 2.12](#), we will assume without further comment that an interpreted word-hyperbolic structure Σ has the property that $A(\Sigma) \subseteq L(\Sigma)$ and that $\alpha(\Sigma)$ is the embedding map from $A(\Sigma)$ to $L(\Sigma)$. Notice further that the computational effectiveness aspect of [Proposition 2.12](#) ensures we are free to assume that an interpreted word-hyperbolic structure serving as input to a decision problem has this property.

For automatic semigroups, it is possible to assume that the automatic structure has a further pleasant property, namely that every element of the semigroup is represented by a unique word in the language of representatives

[KO06, Proposition 2.9(iii)]. However, there exists a word-hyperbolic semi-group (indeed, a word-hyperbolic monoid) that does not admit a word-hyperbolic structure where the language of representatives has this uniqueness property [CM12, Examples 10 & 11].

3 BASIC CALCULATIONS

This section notes a few very basic facts about computing with word-hyperbolic structures for semigroups that are used later in the paper.

LEMMA 3.1 ([HKOT02, Proof of Lemma 3.6]). *There is an algorithm that takes as input a word-hyperbolic structure Σ and two words $p, q \in L(\Sigma)$, and outputs a word $r \in L(\Sigma)$ satisfying $\overline{p} \overline{q} = \overline{r}$ with $|r| \leq c(|p| + |q|)$ (where c is dependent only on Σ) in time $\mathcal{O}((|p| + |q|)^5)$.*

[Actually, the appearance of this lemma in [HKOT02] allows p or q to be empty and asserts that $|r| \leq c(|p| + |q| + 2)$. To obtain the lemma above, where p and q are non-empty, increase c appropriately.] Notice that there may be many possibilities for a word r with $\overline{p} \overline{q} = \overline{r}$.

LEMMA 3.2. *There is an algorithm that takes as input a word-hyperbolic structure Σ and three words $p, q, r \in L(\Sigma)$, and decides whether $\overline{p} \overline{q} = \overline{r}$ in time cubic in $|p| + |q| + |r|$.*

Proof of 3.2. The algorithm simply checks whether $p\#_1 q\#_2 r^{\text{rev}} \in M(\Sigma)$, and the membership problem for context-free languages is soluble in cubic time. 3.2

4 WORD PROBLEM

This section is dedicated to proving that word-hyperbolic semi-groups have word problem soluble in polynomial time.

As noted in the introduction, the previously-known algorithm required exponential time [HKOT02, Theorem 3.8]. This motivated Hoffmann & Thomas to define a narrower notion of word-hyperbolicity for monoids that still generalizes word-hyperbolicity for groups. By restricting to this version of word-hyperbolicity, one recovers automaticity [HT03, Theorem 3] and an algorithm that runs in time $\mathcal{O}(n \log n)$, where n is the length of the input words [HT03, Theorem 2]. Although the algorithm described below is not as fast as this, the existence of a polynomial-time solution to the word problem for word-hyperbolic monoids (in the original Duncan–Gilman sense) diminishes the appeal of the Hoffmann–Thomas restricted version.

THEOREM 4.1. *There is an algorithm that takes as input an interpreted word-hyperbolic structure Σ for a semigroup and two words $w, w' \in A(\Sigma)^+$ and determines whether $\overline{w} = \overline{w'}$ in time polynomial in $|w| + |w'|$.*

Proof of 4.1. By interchanging w and w' if necessary, assume that $|w| \geq |w'|$. First, if $|w| = |w'| = 1$, then $w, w' \in A(\Sigma)$ and so (since the interpretation map is injective on $A(\Sigma)$), we have $\overline{w} = \overline{w'}$ if and only if $w = w'$.

So assume $|w| \geq 2$. Factorize w as $w = w^{(1)}w^{(2)}$, where $w^{(1)} = \lfloor |w|/2 \rfloor$. Notice that $\overline{w} = \overline{w'}$ if and only if $\overline{w^{(1)}} \overline{w^{(2)}} = \overline{w'}$.

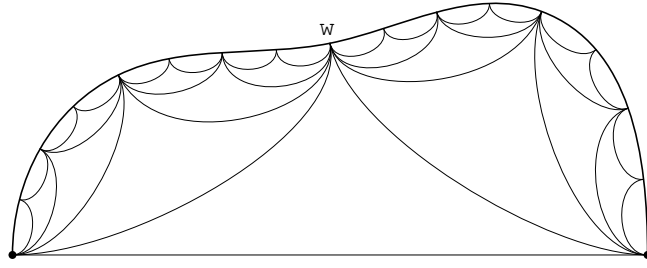


FIGURE 1. Finding a word in R representing the same element as $w \in A^*$.

By Lemma 4.2 below, there is a polynomial-time algorithm that takes the three words $w^{(1)}$, $w^{(2)}$, and w' and yields words $u^{(1)}$, $u^{(2)}$, and u' in $L(\Sigma)$ representing $\overline{w^{(1)}}$, $\overline{w^{(2)}}$, and $\overline{w'}$, of lengths at most $(c+1)|w^{(1)}|^{1+\log(c+1)}$, $(c+1)|w^{(2)}|^{1+\log(c+1)}$ and $(c+1)|w'|^{1+\log(c+1)}$, in time polynomial in $|w^{(1)}|$, $|w^{(2)}|$, and $|w'|$, respectively, where c is a constant dependent only on Σ .

It follows that $\overline{w} = \overline{w'}$ if and only if $\overline{u^{(1)}}\overline{u^{(2)}} = \overline{u'}$, and, by Lemma 3.2, this can be checked in time cubic in $|u^{(1)}| + |u^{(2)}| + |u'|$, which, by the bounds on the lengths of $u^{(1)}$, $u^{(2)}$, and u' , is still polynomial in the lengths of w and w' . Thus the word problem for the semigroup described by Σ is soluble in polynomial time. 4.1

LEMMA 4.2. *There is an algorithm that takes as input an interpreted word-hyperbolic structure Σ for a semigroup and a words $w \in A(\Sigma)^+$ and outputs a word $u \in L(\Sigma)$ with $\overline{w} = \overline{u}$ and $|u| \leq |w|(c+1)|w|^{\log(c+1)}$, and which takes time polynomial in $|w|$.*

Proof of 4.2. Suppose $w = w_1 \cdots w_n$, where $w_i \in A \subseteq L$. Therefore w_1, \dots, w_n is a sequence of words in L whose concatenation represents the same element of the semigroup as w .

For the purposes of this proof, the *total length* of a sequence s_1, \dots, s_ℓ of words in A^* is defined to be the sum of the lengths of the words $|s_1| + \dots + |s_\ell|$.

Consider the following computation, which will form the iterative step of the algorithm: suppose there is a sequence of words s_1, \dots, s_ℓ , each lying in $L(\Sigma)$ and each of length at most t . Notice that ℓt is an upper bound for the total length of this sequence. For $i = 1, \dots, \lfloor \ell/2 \rfloor$, apply Lemma 3.1 to compute a word $s'_i \in L(\Sigma)$ representing $\overline{s_{2i-1}}\overline{s_{2i}}$ of length at most $c(|s_{2i-1}| + |s_{2i}|) \leq 2ct$. For each $i = 1, \dots, \lfloor \ell/2 \rfloor$, this takes $\mathcal{O}((|s_{2i-1}| + |s_{2i}|)^5)$ time, which is at worst $\mathcal{O}((2t)^5)$ time. Therefore the total time used is at most $\mathcal{O}(\lfloor \ell/2 \rfloor (2t)^5)$, which is certainly no worse than time $\mathcal{O}((\ell t)^5)$. That is, the total time used is at worst quintic in the upper bound of the total length of the original sequence.

If ℓ is odd, set $s'_{\lceil \ell/2 \rceil}$ to be s_ℓ . (If ℓ is even, $\lceil \ell/2 \rceil = \lfloor \ell/2 \rfloor$, so $s'_{\lceil \ell/2 \rceil}$ has already been computed.) This is purely notational; no extra computation is done.

The result of this computation is a sequence of $\lceil \ell/2 \rceil$ words, each of length at most $2ct$, whose concatenation represents the same element of the semigroup as the concatenation of the original sequence. The total length of the result is at most $(c+1)\ell t$; that is, at most $c+1$ times the total length of the previous sequence.

Apply this computation iteratively, starting with the sequence w_1, \dots, w_n and continuing until a sequence with only one element results. Since each iteration takes a sequence with ℓ terms to one with $\lceil \ell/2 \rceil$ terms, there are at most $\lceil \log n \rceil$ iterations. The first iteration of this computation, applied to a

sequence whose total length is at most n , completes in time $\mathcal{O}(n^5)$, yielding a sequence of total length at most $n(c+1)$; the next iteration completes in time $\mathcal{O}((n(c+1))^5)$, yielding a sequence of total length at most $n(c+1)^2$. In general the i -th iteration completes in time at most $\mathcal{O}((n(c+1)^{i-1})^5)$, yielding a sequence of total length at most $n(c+1)^i$. So the $\lceil \log n \rceil$ iterations together complete in time at most $\mathcal{O}((1+\log n)(n(c+1)^{1+\log n})^5)$, since $\lceil \log n \rceil \leq 1 + \log n$. [Figure 1 illustrates the algorithm geometrically: each iteration yields a sequence of roughly half as many words in $L(\Sigma)$ labelling a sequence of arcs that each span a subword twice as long as the corresponding terms in the preceding sequence.]

Applying exponent and logarithm laws,

$$\begin{aligned} n(c+1)^{1+\log n} &= n(c+1)(c+1)^{\log n} \\ &= n(c+1)n^{\log(c+1)} \\ &= (c+1)n^{1+\log(c+1)}, \end{aligned}$$

and so, since c is a constant, the algorithm completes in time

$$\mathcal{O}(n^{5+5\log(c+1)} \log n),$$

yielding a word in $L(\Sigma)$ of length at most $n(c+1)n^{\log(c+1)}$. 4.2

Interestingly, although Theorem 4.1 gives a polynomial-time algorithm for the word problem for word-hyperbolic monoids, the proof does not give a bound on the exponent of the polynomial, because the constant c of Lemma 3.1 is dependent on the word-hyperbolic structure Σ . There is thus an open question: does such a bound actually exist? or can the word problem for hyperbolic semigroups be arbitrarily hard within the class of polynomial-time problems?

The algorithm described in Lemma 4.2 is not particularly novel. It is similar in outline to that described by Hoffmann & Thomas [HT03, Lemma 11] for their restricted notion of word-hyperbolicity in monoids. However, the proof that it takes time polynomial in the lengths of the input words is new.

Hoffmann & Thomas describe their algorithm in recursive terms: to find a word in $L(\Sigma)$ representing the same element as $w \in A^*$, factor w as $w'w''$, where the lengths of w' and w'' differ by at most 1, recursively compute representatives p' and p'' in $L(\Sigma)$ of $\overline{w'}$ and $\overline{w''}$, then compute a representative for \overline{w} using p' and p'' . This last step they prove to take linear time (recall that this only applies for their restricted notion of word-hyperbolicity) and to yield a word of length at most $|p'| + |p''| + 1$, which shows that the whole algorithm takes time $\mathcal{O}(n \log n)$. However, this recursive, ‘top-down’ view of the algorithm obscures the fact that the overall strategy can be made to work even for monoids that are word-hyperbolic in the general Duncan–Gilman sense. It is through the iterative, ‘bottom-up’ view of the algorithm presented above that it becomes apparent that the length increase of Lemma 3.1 remains under control through the $\log n$ iterations.

5 DECIDING BASIC PROPERTIES

This section shows that certain basic properties are effectively decidable for word-hyperbolic semigroups. First, being a monoid is decidable:

ALGORITHM 5.1.

Input: An interpreted word-hyperbolic structure Σ for a semigroup.

Output: If the semigroup is a monoid (that is, contains a two-sided identity), output *Yes* and a word in $L(\Sigma)$ representing the identity; otherwise output *No*.

Method:

1. For each $a \in A$, construct the context-free language

$$I_a = \{i \in L : a\#_1 i\#_2 a \in M\} \quad (5.1)$$

and check that it is non-empty. If any of these checks fail, halt and output *No*.

2. For each $a \in A$, choose some $i_a \in I_a$.
3. Iterate the following step for each $a \in A$. For each $b \in B$, if $\overline{i_a} \overline{b} = \overline{b} \overline{i_a} = \overline{b}$, halt and output *Yes* and i_a .
4. Halt and output *No*.

PROPOSITION 5.2. *Algorithm 5.1 outputs Yes and i if and only if the semigroup defined by Σ is a monoid with identity \overline{i} .*

Proof of 5.2. Suppose first that Algorithm 5.1 halts with output *Yes* and i . Then by step 3, $\overline{i} \overline{b} = \overline{b} \overline{i} = \overline{b}$ for all $b \in A$. Since \overline{A} generates S , it follows that $\overline{i} \overline{s} = \overline{s} \overline{i} = \overline{s}$ for all $s \in S$ and hence \overline{i} is an identity for S .

Suppose now that S is a monoid with identity e . Then there is some word $w \in L$ with $\overline{w} = e$. For every $a \in A$, $\overline{a}e = \overline{a}$, and so $a\#_1 w\#_2 a \in M$. Thus $w \in I_a$ for all $a \in A$ and so each I_a is non-empty. Thus the checks in step 1 succeed and the algorithm proceeds to step 2.

Suppose that $w = w_1 \cdots w_n$, where $w_j \in A$ for each $j = 1, \dots, n$. Then

$$\begin{aligned} e &= \overline{w} = \overline{w_1 \cdots w_{n-1} w_n} \\ &= \overline{w_1 \cdots w_{n-1}} \overline{w_n} \overline{i_{w_n}} && \text{(by the choice of } i_{w_n} \in I_{w_n}) \\ &= \overline{e i_{w_n}} \\ &= \overline{i_{w_n}} && \text{(since } e \text{ is an identity for } S). \end{aligned}$$

Hence i_{w_n} represents the identity e and so $\overline{i_{w_n}} \overline{b} = \overline{b} \overline{i_{w_n}} = \overline{b}$. Thus at least one of the i_a chosen in step 2 passes the test of step 3 (which guarantees that it represents an identity since \overline{A} generates S) and so the algorithm halts at step 3 and outputs *Yes* and a word i_a representing the identity. [5.2]

QUESTION 5.3. Is there an algorithm that takes as input an interpreted word-hyperbolic structure and determines whether the semigroup it defines contains a zero?

Notice that this cannot be decided using a procedure like Algorithm 5.1, or at least not obviously, because the natural analogue of I_a is

$$Z_a = \{z \in L : a\#_1 z\#_2 z^{\text{rev}} \in M\},$$

which is naturally defined as the intersection of M and $\{u\#_1 v\#_2 v^{\text{rev}} : u, v \in A^+\}$. However, testing the emptiness of an intersection of context-free languages is in general undecidable. So using Z_a would, at minimum, require some additional insight into the kind of context-free languages that can appear as M .

Notice that commutativity is very easy to decide for a word-hyperbolic semigroup; one needs to check only that $\overline{ab} = \overline{ba}$ for all symbols $a, b \in A(\Sigma)$. This is simply a matter of performing a bounded number of multiplications and checks using [Lemmata 3.1](#) and [3.2](#).

Green's relation \mathcal{L} is decidable for automatic semigroups; in contrast, Green's relation \mathcal{R} is undecidable, as a corollary of the fact that right-invertibility is undecidable in automatic monoids [[KO06](#), Theorem 5.1]. In contrast, \mathcal{R} and \mathcal{L} are both decidable for word-hyperbolic semigroups, as a consequence of $M(\Sigma)$ describing the entire multiplication table.

PROPOSITION 5.4. *There is an algorithm that takes as input an interpreted word-hyperbolic structure Σ and two words $w, w' \in L(\Sigma)$ and decides whether the elements represented by w and w' are:*

1. \mathcal{R} -related,
2. \mathcal{L} -related,
3. \mathcal{H} -related.

Proof of 5.4. Let S be the semigroup described by Σ . The elements \overline{w} and $\overline{w'}$ are \mathcal{R} -related if and only if there exist $s, t \in S^1$ such that $\overline{w}s = \overline{w'}$ and $\overline{w'}t = \overline{w}$. That is, $\overline{w} \mathcal{R} \overline{w'}$ if and only if either $\overline{w} = \overline{w'}$, or there exist $s, t \in S$ with $\overline{w}s = \overline{w'}$ and $\overline{w'}t = \overline{w}$. The possibility that $\overline{w} = \overline{w'}$ can be checked algorithmically by [Theorem 4.1](#). The existence of an element $s \in S$ such that $\overline{w}s = \overline{w'}$ is equivalent to the non-emptiness of the language

$$\{v \in L : w\#_1 v\#_2 (w')^{\text{rev}} \in M\}.$$

This context-free language can be effectively constructed and its non-emptiness effectively decided. Similarly, it is possible to decide whether there is an element $t \in S$ such that $\overline{w'}t = \overline{w}$. Hence it is possible to decide whether $\overline{w} \mathcal{R} \overline{w'}$.

Similarly, one can effectively decide whether $\overline{w} \mathcal{L} \overline{w'}$. Since $\overline{w} \mathcal{H} \overline{w'}$ if and only if $\overline{w} \mathcal{R} \overline{w'}$ and $\overline{w} \mathcal{L} \overline{w'}$, whether \overline{w} and $\overline{w'}$ are \mathcal{H} -related is effectively decidable. [5.4]

COROLLARY 5.5. *There is an algorithm that takes as input an interpreted word-hyperbolic structure and decides whether the semigroup it describes is a group.*

Proof of 5.5. Suppose the input word-hyperbolic structure is Σ and that it describes a semigroup S . Apply [Algorithm 5.1](#). If S is not a monoid, it cannot be a group. Otherwise we know that S is a monoid and we have a word $\bar{i} \in L(\Sigma)$ that represents its identity. For each $a \in A(\Sigma)$, check whether $\overline{a} \mathcal{R} \bar{i}$ and $\overline{a} \mathcal{L} \bar{i}$: if all these checks succeed, then every generator is both right- and left-invertible, and so S is a group; if any fail, there is some generator that is either not right- or not left-invertible and so S cannot be a group. Hence it is decidable whether Σ describes a group. [5.5]

QUESTION 5.6. Are Green's relations \mathcal{D} and \mathcal{J} decidable for word-hyperbolic semigroups?

Note that \mathcal{D} and \mathcal{J} are both undecidable for automatic semigroups [[Otto7](#), Theorems 4.1 & 4.3].

This section shows that it is decidable whether a word-hyperbolic semigroup is completely simple. This is particularly useful because a completely simple semigroup is word-hyperbolic if and only if its Cayley graph is a hyperbolic metric space [FK04, Theorem 4.1], generalizing the equivalence for groups of these properties for groups.

DEFINITION 6.1. Let S be a semigroup, I and Λ be index sets, and P be a $\Lambda \times I$ matrix over S whose (λ, i) -th element is $p_{\lambda, i}$. The Rees matrix semigroup $\mathcal{M}[S; I, \Lambda; P]$ is defined to be the set $I \times S \times \Lambda$ with multiplication

$$(i, g, \lambda)(j, h, \mu) = (i, gp_{\lambda, j}h, \mu).$$

Recall that a semigroup is completely simple if it has no proper two-sided ideals, is not the two-element null semigroup, and contains a primitive idempotent (that is, an idempotent e such that, for all idempotents f , we have $ef = fe = f \implies e = f$). The version of the celebrated Rees theorem due to Suschkewitsch [How95, Theorem 3.3.1] shows that all completely simple semigroups are isomorphic to a semigroup $\mathcal{M}[G; I, \Lambda; P]$, where G is a group and I and Λ are finite sets.

Let A be an alphabet representing a generating set for a completely simple semigroup $\mathcal{M}[G; I, \Lambda; P]$. Define maps $\nu : A \rightarrow I$ and $\xi : A \rightarrow \Lambda$ by letting $a\nu$ and $a\xi$ be such that $\bar{a} \in \{a\nu\} \times G \times \{a\xi\}$. For the purposes of this paper, we call the pair of maps (ν, ξ) the *species* of the completely simple semigroup. We first of all prove that it is decidable whether a word-hyperbolic semigroup is a completely simple semigroup of a particular species.

ALGORITHM 6.2.

Input: An interpreted word-hyperbolic structure Σ , two finite sets I and Λ , and two surjective maps $\nu : A(\Sigma) \rightarrow I$ and $\xi : A(\Sigma) \rightarrow \Lambda$.

Output: If Σ describes a completely simple semigroup of species (ν, ξ) , output *Yes*; otherwise output *No*.

Method: At various points in the algorithm, checks are made. If any of these checks fail, the algorithm halts and outputs *No*.

1. For each $i \in I$ and $\lambda \in \Lambda$, construct the regular language

$$L_{i, \lambda} = \{a_1 \cdots a_n \in L : a_i \in A, a_1\nu = i, a_n\xi = \lambda\}.$$

Check that each $L_{i, \lambda}$ is non-empty.

2. For each $i, j \in I$ and $\lambda, \mu \in \Lambda$, construct the context-free language

$$\{u\#_1\nu\#_2w^{\text{rev}} \in M : u \in L_{i, \lambda}, \nu \in L_{j, \mu}, w \in L - L_{i, \mu}\}, \quad (6.1)$$

and check that it is empty.

3. For each $i \in I$ and $\lambda \in \Lambda$, choose a word $w_{i, \lambda} \in L_{i, \lambda}$ and construct the context-free language

$$I_{i, \lambda} = \{u \in L_{i, \lambda} : w_{i, \lambda}\#_1u\#_2w_{i, \lambda}^{\text{rev}} \in M\}.$$

Check that each $I_{i, \lambda}$ is non-empty.

4. For each $i \in I$ and $\lambda \in \Lambda$, choose a word $u_{i, \lambda} \in I_{i, \lambda}$.
5. For each $a \in A$, $i \in I$, and $\lambda \in \Lambda$, check that $\overline{u_{a\nu, \lambda}}\bar{a} = \bar{a}$ and $\bar{a}\overline{u_{i, a\xi}} = \bar{a}$.

6. For each $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$, calculate a word $h_{i,a,\mu,\lambda} \in L$ such that $\overline{h_{i,a,\mu,\lambda}} = \overline{u_{i,\mu}} \overline{a} \overline{u_{i,\lambda}}$.
7. For each $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$, check that $\overline{h_{i,a,\mu,\lambda}} \overline{u_{i,a\xi}} = \overline{u_{i,\mu}} \overline{a}$.
8. For each $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$, check that

$$\overline{u_{i,\lambda}} \overline{h_{i,a,\mu,\lambda}} = \overline{h_{i,a,\mu,\lambda}} \overline{u_{i,\lambda}} = \overline{h_{i,a,\mu,\lambda}}.$$

9. For each $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$, construct the context-free language

$$V_{i,a,\mu,\lambda} = \{v \in L : h_{i,a,\mu,\lambda} \#_1 v \#_2 u_{i,\lambda} \in M\}$$

and check that it is non-empty.

10. For each $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$, choose some $v_{i,a,\mu,\lambda} \in V_{i,a,\mu,\lambda}$ and check that $\overline{a} \overline{h_{i,a,\mu,\lambda}} = \overline{u_{i,\lambda}}$.
11. Halt and output Yes.

Lemmata 6.3 and 6.4 show that this algorithm works.

LEMMA 6.3. *If Algorithm 6.2 outputs Yes, the semigroup defined by the word-hyperbolic structure Σ is a completely simple semigroup of species (v, ξ) .*

Proof of 6.3. Let S be the semigroup defined by the input word-hyperbolic structure Σ . Suppose the algorithm output Yes. Then all the checks in steps 1–10 must succeed.

For each $i \in I$ and $\lambda \in \Lambda$, let $T_{i,\lambda} = \overline{L_{i,\lambda}}$. By the definition of $L_{i,\lambda}$, for each $a \in A$, the word a lies in $L_{av,a\lambda}$. By the check in step 1, each $T_{i,\lambda}$ is non-empty.

By the check in step 2, for all $i, j \in I$ and $\lambda, \mu \in \Lambda$, there do not exist $u \in L_{i,\lambda}$, $v \in L_{j,\mu}$, $w \in L - L_{i,\mu}$ with $\overline{u} \overline{v} = \overline{w}$. That is,

$$T_{i,\lambda} T_{j,\mu} \subseteq T_{i,\mu} \text{ for all } i, j \in I \text{ and } \lambda, \mu \in \Lambda. \quad (6.2)$$

In particular, $T_{i,\lambda} T_{i,\lambda} \subseteq T_{i,\lambda}$ and so each $T_{i,\lambda}$ is a subsemigroup of S .

In each $T_{i,\lambda}$, there is some element that stabilizes some other element $\overline{w_{i,\lambda}}$ on the right (that is, that right-multiplies $\overline{w_{i,\lambda}}$ like an identity) by the check in step 3. In step 4, $u_{i,\lambda}$ is chosen to be such an element. Let $e_{i,\lambda} = \overline{u_{i,\lambda}}$.

By the check in step 5,

$$e_{av,\lambda} \overline{a} = \overline{a} \text{ and } \overline{a} e_{i,a\xi} = \overline{a} \text{ for all } i \in I \text{ and } \lambda \in \Lambda. \quad (6.3)$$

In step 6, $h_{i,a,\mu,\lambda}$ is calculated for all $i \in I$, $\lambda, \mu \in \Lambda$, $a \in A$ so that

$$\overline{h_{i,a,\mu,\lambda}} = e_{i,\mu} \overline{a} e_{i,\lambda}. \quad (6.4)$$

By (6.2), $h_{i,a,\mu,\lambda} \in L_{i,\lambda}$. By the check in step 7,

$$\overline{h_{i,a,\mu,\lambda}} e_{i,a\xi} = e_{i,\mu} \overline{a} \text{ for all } i \in I, \lambda, \mu \in \Lambda, a \in A. \quad (6.5)$$

Let $i \in I$ and $\lambda \in \Lambda$. Let $t \in T_{i,\lambda}$. Then $t = \overline{a_1 a_2 \cdots a_n}$ for some $a_k \in A$.

Since $a_1 a_2 \cdots a_n \in L_{i,\lambda}$, $a_1 v = i$ and $a_n \xi = \lambda$. Then

$$\begin{aligned}
& \overline{a_1 a_2 a_3 \cdots a_n} \\
&= e_{i,\lambda} \overline{a_1 a_2 a_3 \cdots a_n} e_{i,\lambda} && [\text{by (6.3), since } a_1 v = i \text{ and } a_n \xi = \lambda] \\
&= \overline{h_{i,a_1,\lambda,\lambda} e_{i,a_1 \xi} \overline{a_2 a_3 \cdots a_n} e_{i,\lambda}} && [\text{by (6.5)}] \\
&= \overline{h_{i,a_1,\lambda,\lambda} \overline{h_{i,a_2,a_1 \xi,\lambda} e_{i,a_2 \xi} \overline{a_3 \cdots a_n} e_{i,\lambda}}} && [\text{by (6.5)}] \\
&= \overline{h_{i,a_1,\lambda,\lambda} \overline{h_{i,a_2,a_1 \xi,\lambda} \overline{h_{i,a_3,a_2 \xi,\lambda} e_{i,a_3 \xi} \cdots \overline{a_n} e_{i,\lambda}}}} && [\text{by (6.5)}] \\
&\quad \vdots \\
&= \overline{h_{i,a_1,\lambda,\lambda} \overline{h_{i,a_2,a_1 \xi,\lambda} \overline{h_{i,a_3,a_2 \xi,\lambda} \cdots e_{i,a_{n-1} \xi} \overline{a_n} e_{i,\lambda}}}} && [\text{by repeated use of (6.5)}] \\
&= \overline{h_{i,a_1,\lambda,\lambda} \overline{h_{i,a_2,a_1 \xi,\lambda} \overline{h_{i,a_3,a_2 \xi,\lambda} \cdots h_{i,a_n,a_{n-1} \xi,\lambda}}}} && [\text{by (6.4)}]
\end{aligned}$$

Therefore the subsemigroup $T_{i,\lambda}$ is generated by the set of elements $H_{i,\lambda} = \{\overline{h_{i,a,\mu,\lambda}} : a \in A, \mu \in \Lambda\}$.

By the check in step 8, for all $i \in I$, $\lambda \in \Lambda$, and $h \in H_{i,\lambda}$, we have $h e_{i,\lambda} = e_{i,\lambda} h = h$. Since $H_{i,\lambda}$ generates $T_{i,\lambda}$, it follows that $e_{i,\lambda}$ is an identity for $T_{i,\lambda}$. So each $T_{i,\lambda}$ is a submonoid of S with identity $e_{i,\lambda}$. In particular, each $e_{i,\lambda}$ is idempotent.

Let $i \in I$ and $\lambda \in \Lambda$. By the check in step 9, every element $h \in H_{i,\lambda}$ has a right inverse h' in $T_{i,\lambda}$. By the check in step 10, $h'h = e_{i,\lambda}$ and so h' is also a left-inverse for h in $T_{i,\lambda}$. Thus every generator in $H_{i,\lambda}$ is both right- and left-invertible. Hence every element of $T_{i,\lambda}$ is both right- and left-invertible and so $T_{i,\lambda}$ is a subgroup of S .

Since S is the union of the various $T_{i,\lambda}$, the semigroup S is regular and the $e_{i,\lambda}$ are the only idempotents in S . Thus by (6.2), distinct idempotents cannot be related by the idempotent ordering. Hence all idempotents of S are primitive. Since S does not contain a zero (since it is the union of the $T_{i,\lambda}$ and (6.2) holds), it is completely simple by [How95, Theorem 3.3.3]. 6.3

LEMMA 6.4. *If semigroup defined by the word-hyperbolic structure Σ is a completely simple semigroup of species (v, ξ) , then Algorithm 6.2 outputs Yes.*

Proof of 6.4. Suppose the semigroup S defined by the word-hyperbolic structure Σ is a completely simple semigroup, with $S = \mathcal{M}[G; I, \Lambda; P]$. For all $i \in I$ and $\lambda \in \Lambda$, let $e_{i,\lambda}$ be the identity of the subgroup $T_{i,\lambda} = \{i\} \times G \times \{\lambda\}$; that is, $e_{i,\lambda} = (i, p_{\lambda,i}^{-1}, \lambda)$. For each $a \in A$, the element \overline{a} has the form $(av, g_a, a\xi)$ for some $g_a \in G$.

By the definition of multiplication in S , the word $a_1 \cdots a_n \in L$ represents an element of $T_{i,\lambda}$ if and only if $a_1 v = i$ and $a_n \xi = \lambda$. Hence each $L_{i,\lambda}$ must be the preimage of $T_{i,\lambda}$ and map surjectively onto $T_{i,\lambda}$. In particular, $L_{i,\lambda}$ must be non-empty and so the checks in step 1 succeed.

For any $i, j \in I$ and $\lambda, \mu \in \Lambda$, we have $T_{i,\lambda} T_{j,\mu} \subseteq T_{i,\mu}$. Hence if $u \in L_{i,\lambda}$, $v \in L_{j,\mu}$, and $w \in L$ are such that $\overline{uv} = \overline{w}$, then $w \in L_{i,\mu}$. Thus the language (6.1) is empty for all $i, j \in I$ and $\lambda, \mu \in \Lambda$. Hence all the checks in step 2 succeed.

For any $i \in I$ and $\lambda \in \Lambda$, if $w_{i,\lambda} \in L_{i,\lambda}$, then $\overline{w_{i,\lambda}} \in T_{i,\lambda}$. Since $T_{i,\lambda}$ is a subgroup, $\overline{w_{i,\lambda}} e_{i,\lambda} = \overline{w_{i,\lambda}}$, and $e_{i,\lambda}$ is the unique element of $T_{i,\lambda}$ that stabilizes $\overline{w_{i,\lambda}}$ on the right. Thus the language $I_{i,\lambda}$ is non-empty, and consists of words representing $e_{i,\lambda}$. Hence the checks in step 3 succeed, and the words $u_{i,\lambda}$ chosen in step 4 are such that $\overline{u_{i,\lambda}} = e_{i,\lambda}$.

In a completely simple semigroup, each idempotent is a left identity within its own \mathcal{R} -class and $e_{i,a\xi}$ is a right identity within its own \mathcal{L} -class [How95, Proposition 2.3.3]. Hence for each $a \in A$, $i \in I$, and $\lambda \in \Lambda$, we have $e_{av,\lambda}\bar{a} = \bar{a}$ and $\bar{a}e_{i,a\xi} = \bar{a}$. Thus the checks in step 5 succeed.

For all $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$,

$$\begin{aligned} & \overline{h_{i,a,\mu,\lambda} u_{i,a\xi}} \\ &= e_{i,\mu} \bar{a} e_{i,\lambda} e_{i,a\xi} \\ &= e_{i,\mu} (av, g_a, a\xi) (i, p_{\lambda,i}^{-1}, \lambda) (i, p_{a\xi,i}^{-1}, a\xi) \\ &= e_{i,\mu} (av, g_a p_{a\xi,i} p_{\lambda,i}^{-1} p_{\lambda,i}^{-1} p_{a\xi,i}^{-1}, a\xi) \\ &= e_{i,\mu} (av, g_a, a\xi) \\ &= e_{i,\mu} \bar{a}. \end{aligned}$$

Thus all the checks in step 7 succeed.

For all $a \in A$, $i \in I$, and $\lambda, \mu \in \Lambda$, the element $\overline{h_{i,a,\mu,\lambda}}$ lies in the subgroup $T_{i,\lambda}$, whose identity is $e_{i,\lambda}$. Hence all the checks in step 8 succeed. Since all elements of this subgroup are right-invertible, each language $V_{i,a,\mu,\lambda}$ is non-empty; hence all the checks in step 9 succeed. Finally, since a right inverse is also a left inverse in a group, all the checks in step 10 succeed. Therefore the algorithm reaches step 10 and halts with output *Yes*. 6.4

THEOREM 6.5. *There is an algorithm that takes as input an interpreted word-hyperbolic structure Σ for a semigroup and decides whether it is a completely simple semigroup.*

Proof of 6.5. We prove that this problem can be reduced to the problem of deciding whether the semigroup defined by an interpreted word-hyperbolic structure Σ is a completely simple semigroup of a particular species ($v : A(\Sigma) \rightarrow I, \xi : A(\Sigma) \rightarrow \Lambda$).

Let S be the semigroup specified by Σ . Then S is finitely generated. Thus we need only consider the problem of deciding whether S is a finitely generated completely simple semigroup. By the definition of multiplication in a completely simple semigroup (viewed as a Rees matrix semigroup), the leftmost generator in a product determines its \mathcal{R} -class (that is, the I -component of the product) and the rightmost generator in a product determines its \mathcal{L} -class (that is, the Λ -component of the product). Thus there must be at least one generator in each \mathcal{R} - and \mathcal{L} -class, and hence if S is an $I \times \Lambda$ Rees matrix semigroup, both $|I|$ and $|\Lambda|$ cannot exceed $|A(\Sigma)|$.

Thus it suffices to decide whether S is an $I \times \Lambda$ completely simple semigroup for some fixed choice of I and Λ , for one can simply test the finitely many possibilities for index sets I and Λ no larger than $A(\Sigma)$.

One can restrict further, and ask whether S is completely semigroup of some particular species ($v : A(\Sigma) \rightarrow I, \xi : A(\Sigma) \rightarrow \Lambda$), for there are a bounded number of possibilities for the maps surjective v and ξ , so it suffices to test each one. 6.5

7 BEING A CLIFFORD SEMIGROUP

This section is dedicated to showing that being a Clifford semigroup is decidable for word-hyperbolic semigroups. Recall the definition of a Clifford semigroup:

DEFINITION 7.1. Let Y be a [meet] semilattice and let $\{G_\alpha : \alpha \in Y\}$ be a collection of disjoint groups with, for all $\alpha, \beta \in Y$ such that $\alpha \geq \beta$, a homomorphism $\phi_{\alpha,\beta} : G_\alpha \rightarrow G_\beta$ satisfying the following conditions:

1. For each $\alpha \in Y$, the homomorphism $\phi_{\alpha,\alpha}$ is trivial.
2. For $\alpha, \beta, \gamma \in Y$ with $\alpha \geq \beta \geq \gamma$,

$$\phi_{\alpha,\gamma} = \phi_{\alpha,\beta} \phi_{\beta,\gamma}. \quad (7.1)$$

The set of elements of the *Clifford semigroup* $\mathcal{S}[Y; G_\alpha; \phi_{\alpha,\beta}]$ is the union of the disjoint groups G_α . The product of the elements s and t of S , where $s \in G_\alpha$ and $t \in G_\beta$, is

$$(s\phi_{\alpha,\alpha \wedge \beta})(t\phi_{\beta,\alpha \wedge \beta}), \quad (7.2)$$

which lies in the group $G_{\alpha \wedge \beta}$. [The meet of α and β is denoted $\alpha \wedge \beta$.]

Let A be an alphabet representing a generating set for a Clifford semigroup $\mathcal{S}[Y; G_\alpha; \phi_{\alpha,\beta}]$. Define a map $\xi : A \rightarrow Y$ by letting $a\xi$ be such that $\bar{a} \in G_{a\xi}$. For the purposes of this paper, we call this map $\xi : A \rightarrow Y$ the *species* of the Clifford semigroup. [Notice that the map ξ extends to a unique homomorphism $\xi : A^+ \rightarrow Y$.] We first of all prove that it is decidable whether a word-hyperbolic semigroup is a Clifford semigroup of a particular species.

ALGORITHM 7.2.

Input: An interpreted word-hyperbolic structure Σ and a map $\xi : A \rightarrow Y$.

Output: If Σ describes a Clifford semigroup of species $\xi : A \rightarrow Y$, output *Yes*; otherwise output *No*.

Method: At various points in the algorithm, checks are made. If any of these checks fail, the algorithm halts and outputs *No*.

1. For each $\alpha \in Y$, construct the regular language

$$L_\alpha = \{w \in L : w\xi = \alpha\}.$$

(These languages are regular since L is regular, Y is finite, and the map $\xi : A \rightarrow Y$ is known.) Check that each L_α is non-empty.

2. For each $\alpha, \beta \in Y$, construct the context-free language

$$\{u\#_1 v\#_2 w^{\text{rev}} \in M : u \in L_\alpha, v \in L_\beta, w \in L - L_{\alpha \wedge \beta}\} \quad (7.3)$$

and check that it is empty.

3. For each $\alpha \in Y$, choose some word $w_\alpha \in L_\alpha$ and construct the context-free language

$$I_\alpha = \{i \in L_\alpha : w_\alpha \#_1 i\#_2 w_\alpha^{\text{rev}} \in M\}$$

and check that I_α is non-empty.

4. For each $\alpha \in Y$, pick some $i_\alpha \in I_\alpha$ and check that for all $\alpha, \beta \in Y$, $\overline{i_\alpha i_\beta} = \overline{i_{\alpha \wedge \beta}}$.

5. For each $a \in A$, check that $\overline{i_{a\xi} \bar{a}} = \bar{a} \overline{i_{a\xi}} = \bar{a}$. For each $\alpha \in Y$ and $a \in A$ check that $\bar{a} \overline{i_\alpha} = \overline{i_\alpha} \bar{a}$.

6. For each $\alpha \in Y$ and $a \in A$ such that $a\xi \geq \alpha$, construct the context-free language

$$V_{\alpha,a} = \{v \in L_\alpha : a\#_1 v\#_2 i_\alpha \in M\}$$

and check that $V_{\alpha,a}$ is non-empty.

7. For each $\alpha \in Y$ and $a \in A$ such that $a\xi \geq \alpha$, pick some $v_{\alpha,a} \in V_{\alpha,a}$ and check that $\overline{v_{\alpha,a}} \bar{a} = \bar{i}_\alpha$.
8. Halt and output Yes.

Lemmata 7.3 and 7.4 show that this algorithm works.

LEMMA 7.3. *If Algorithm 7.2 outputs Yes, the semigroup described by the word-hyperbolic structure Σ is a Clifford semigroup of species $\xi : A \rightarrow Y$.*

Proof of 7.3. Let S be the semigroup defined by the input word-hyperbolic structure Σ . Suppose the algorithm output Yes. Then all the checks in steps 1–7 must succeed.

For each $\alpha \in Y$, let $T_\alpha = \overline{L_\alpha}$. By the check in step 1, all T_α are non-empty.

By the check in step 2, for every $\alpha, \beta \in Y$, there do not exist $u \in L_\alpha, v \in L_\beta, w \in L - L_{\alpha \wedge \beta}$ with $\bar{u}\bar{v} = \bar{w}$. That is, $T_\alpha T_\beta \subseteq T_{\alpha \wedge \beta}$. In particular, $T_\alpha T_\alpha \subseteq T_\alpha$ and so each T_α is a subsemigroup of S .

In each T_α , there is some element that right-multiplies some other element like an identity by the check in step 3.

For each $\alpha \in Y$, the word i_α represents an element e_α , and the set of elements $E = \{e_\alpha : \alpha \in Y\}$ forms a subsemigroup isomorphic to the semilattice Y by the check in step 4.

By the checks in step 5, for each $a \in A$, the element $e_{a\xi}$ (which, like \bar{a} , lies in $T_{a\xi}$) acts like an identity on \bar{a} (that is, $e_{a\xi} \bar{a} = \bar{a} e_{a\xi} = \bar{a}$), and every element e_α commutes with \bar{a} .

Let $\alpha \in Y$ and $t \in T_\alpha$. Then $t = \bar{a}_1 \bar{a}_2 \cdots \bar{a}_n$ for some $a_i \in A$ with $(a_1 a_2 \cdots a_n)\xi = \alpha$. Then

$$\begin{aligned}
& \bar{a}_1 \bar{a}_2 \cdots \bar{a}_n \\
&= e_{a_1 \xi} \bar{a}_1 e_{a_2 \xi} \bar{a}_2 \cdots e_{a_n \xi} \bar{a}_n && \text{[by the check in step 6]} \\
&= e_{a_1 \xi} e_{a_2 \xi} \cdots e_{a_n \xi} \bar{a}_1 \bar{a}_2 \cdots \bar{a}_n && \text{[by the check in step 6]} \\
&= e_{(a_1 \xi) \wedge (a_2 \xi) \wedge \cdots \wedge (a_n \xi)} \bar{a}_1 \bar{a}_2 \cdots \bar{a}_n && \text{[by the isomorphism of } E \text{ and } Y] \\
&= e_{(a_1 a_2 \cdots a_n) \xi} \alpha \bar{a}_1 \bar{a}_2 \cdots \bar{a}_n && \text{[by the extension of } \xi \text{ to } A^+] \\
&= e_\alpha \bar{a}_1 \bar{a}_2 \cdots \bar{a}_n.
\end{aligned}$$

Thus $t = e_\alpha t$. Similarly $t e_\alpha = t$. Hence e_α is an identity for T_α .

For each $\alpha \in Y$ and $a \in A$ with $a\xi \geq \alpha$, there is an element $\bar{v}_{\alpha,a} \in T_\alpha$ such that $\overline{v_{\alpha,a}} \bar{a} = \bar{a} \bar{v}_{\alpha,a} = e_\alpha$ by the checks in steps 6 and 7. Since T_α is generated by elements \bar{a} such that $a\xi \geq \alpha$, it follows that T_α is a subgroup of S .

Since L is the union of the various L_α , the semigroup S is the union of the various subgroups T_α . In particular, S is regular. Furthermore, the only idempotents in S are the identities of these subgroups; that is, the elements e_α . Since every e_α commutes with every element of \bar{A} , it follows that all idempotents of S are central. Hence S is a regular semigroup in which the idempotents are central, and thus is a Clifford semigroup by [How95, Theorem 4.2.1]. [7.3]

LEMMA 7.4. *If the semigroup defined by the word-hyperbolic structure Σ is a Clifford semigroup of species $\xi : A \rightarrow Y$, then Algorithm 7.2 outputs Yes.*

Proof of 7.4. Suppose the semigroup S defined by the word-hyperbolic structure $(A, L, M(L))$ is a Clifford semigroup, with $S = \mathcal{S}[Y; G_\alpha; \phi_{\alpha,\beta}]$. For each $\alpha \in Y$, let e_α be the identity of G_α . The language L_α clearly consists of exactly

those words in L that map onto G_α , so L_α is non-empty. Hence the checks in step 1 succeed.

By the definition of multiplication in a Clifford semigroup, $G_\alpha G_\beta \subseteq G_{\alpha \wedge \beta}$. Hence if $u \in L_\alpha$, $v \in L_\beta$, and $w \in L$ are such that $\bar{u}\bar{v} = \bar{w}$, then $w \in L_{\alpha \wedge \beta}$. Thus the language (7.3) is empty for all $\alpha, \beta \in Y$. Hence all the checks in step 2 succeed.

Let $\alpha \in Y$. For any $w_\alpha \in L_\alpha$, the element \bar{w}_α lies in the subgroup G_α . Thus the language I_α consists of precisely the words that represent elements of G_α that right-multiply w_α like an identity. Since G_α is a subgroup, every element of I_α represents e_α . Since there must be at least one such representative, I_α is non-empty. Thus every check in step 3 succeeds.

The identities e_α form a subsemigroup isomorphic to the semilattice Y by the definition of multiplication in a Clifford semigroup. Thus every check in step 4 succeeds.

Furthermore, every e_α is idempotent and thus central in S by [How95, Theorem 4.2.1], and so every check in step 5 succeeds.

Let $\alpha \in Y$ and $a \in A$ be such that $a\xi \geq \alpha$. Let $v_{\alpha,a}$ be the word representing $(\bar{a}\phi_{a\xi,\alpha})^{-1}$. Then

$$\bar{u}_a \bar{v}_{\alpha,a} = \bar{a}(\bar{a}\phi_{a\xi,\alpha})^{-1} = (\bar{a}\phi_{a\xi,\alpha})(\bar{a}\phi_{a\xi,\alpha})^{-1} = e_\alpha.$$

Hence $v_{\alpha,a} \in V_{\alpha,a}$ and so all the checks in step 6 succeed. Similarly $\bar{v}_{\alpha,a} \bar{u}_a$ and so all the checks in step 7 succeed.

Therefore the algorithm reaches step 8 and halts with output *Yes*. [7.4]

THEOREM 7.5. *There is an algorithm that takes as input an interpreted word-hyperbolic structure Σ for a semigroup and decides whether it is a Clifford semigroup.*

Proof of 7.5. We prove that this problem can be reduced to the problem of deciding whether the semigroup defined by an interpreted word-hyperbolic structure Σ is a Clifford semigroup with a particular species $\xi : A(\Sigma) \rightarrow Y$.

Let S be the semigroup specified by Σ . Then S is finitely generated. Thus we need only consider the problem of deciding whether S is a finitely generated Clifford semigroup, whose corresponding semilattice must therefore also be finitely generated. A finitely generated semilattice is finite.

So if S is a Clifford semigroup $\mathcal{S}[Y; G_\alpha; \phi_{\alpha,\beta}]$, the semilattice Y must be a homomorphic image of the free semilattice of rank $|A(\Sigma)|$, which has $2^{|A(\Sigma)|} - 1$ elements. Thus it suffices to decide whether S is a Clifford semigroup for some fixed semilattice Y , for one can simply test the finitely many possibilities for Y .

One can restrict further, and ask whether S is a Clifford semigroup with some fixed semilattice Y and some particular placement of generators into the semilattice of groups. (That is, with knowledge of in which group G_α each generator \bar{a} putatively lies, described by a map $\xi : A(\Sigma) \rightarrow Y$. Of course, it is necessary that $\text{im } \xi$ generates Y .) There are a bounded number of possibilities for the map ξ , so it suffices to test each one. [7.5]

8 BEING FREE

This section shows that it is decidable whether a word-hyperbolic semigroup is free. The following technical lemma, which is possibly of independent interest, is necessary.

LEMMA 8.1. *There is an algorithm that takes as input an alphabet A , a symbol $\#_2$ not in A , and a context-free grammar Γ defining a context-free language $L(\Gamma)$ that is a subset of $A^*\#_2A^*$, and decides whether $L(\Gamma)$ contains a word $x\#_2w^{\text{rev}}$ where $x \neq w$.*

Proof of 8.1. Suppose $\Gamma = (N, A \cup \{\#_2\}, P, O)$. [Here, N is the set of non-terminal symbols, $A \cup \{\#_2\}$ is of course the set of terminal symbols, P the set of productions, and $O \in N$ is the start symbol.] Since M does not contain the empty word (since every word in M lies in $A^*\#_2A^*$), assume without loss that Γ contains no useless symbols or unit productions [HU79, Theorem 4.4].

Let

$$N_{\#} = \{M \in N : (\exists u, v \in A^*)(M \Rightarrow^* p\#_2q)\}.$$

Notice that if $M \rightarrow p$ is a production in P and $M \in N - N_{\#}$, then every non-terminal symbol appearing in p also lies in $N - N_{\#}$. [This relies on there being no useless symbols in Γ , which means that every other non-terminal in P derives some terminal word.] For this reason, it is easy to compute $N_{\#}$.

Suppose that $M \Rightarrow^* uMv$ for some $M \in N - N_{\#}$ and $u, v \in (A \cup \{\#_2\})^*$. Then u and v cannot contain $\#_2$ since $M \in N - N_{\#}$. Since there are no unit productions in P , at least one of u and v is not the empty word. Since M is not a useless symbol, it appears in some derivation of a word $w\#_2x^{\text{rev}} \in L(\Gamma)$. Pumping the derivation $M \Rightarrow^* uMv$ yields a word $w'\#_2(x')^{\text{rev}}$ where exactly one of $w' = w$ or $x' = x$ holds, since the extra inserted u and v cannot be on opposite sides of the symbol $\#_2$ since $M \in N - N_{\#}$. Hence either $w \neq x$ or $w' \neq x'$. Hence in this case $L(\Gamma)$ does contain a word of the given form.

Since it is easy to check whether there is a non-terminal $M \in N - N_{\#}$ with $M \Rightarrow^* uMv$, we can assume that no such non-terminal exists. Therefore any non-terminal $M \in N - N_{\#}$ derives only finitely many words (since any derivation starting at M can only involve non-terminals in $N - N_{\#}$ and by assumption no such non-terminal can appear twice in a given derivation). These words can be effectively enumerated. Let $M \in N - N_{\#}$ and let w_1, \dots, w_n be all the words that M derives. Replacing a production $S \rightarrow pMq$ by the productions $S \rightarrow pw_1q, S \rightarrow pw_2q, \dots, S \rightarrow pw_nq$ does not alter $L(\Gamma)$. Iterating this process, we eventually obtain a grammar Γ where no non-terminal symbol in $N - N_{\#}$ appears on the right-hand side of a production. Thus all symbols in $N - N_{\#}$ can be eliminated and we now have a grammar Γ with $N = N_{\#}$.

Every production is now of the form $M \rightarrow pSq$ or $M \rightarrow p\#_2q$, where $p, q \in A^*$ and $S \in N$. [There can be only one non-terminal on the right-hand side of each production, since otherwise some terminal word would contain two symbols $\#_2$, which is impossible.]

We are now going to iteratively define a map $\phi : N \rightarrow FG(A)$, where $FG(A)$ denotes the free group on A , which we will identify with the set of reduced words on $A \cup A^{-1}$. First, define $O\phi = \varepsilon$. Now, iterate through the productions as follows. Choose some production $M \rightarrow pSq^{\text{rev}}$ such that $M\phi$ is already defined. Let $z = p^{-1}(M\phi)q \in FG(A)$. If $S\phi$ is undefined, set $S\phi = z$. If $S\phi$ is defined, check that $S\phi$ and z are equal; if they are not, halt: $L(\Gamma)$ does contain words $w\#_2x^{\text{rev}}$ with $w \neq x$.

To see this, suppose $S\phi = z$ and consider the sequence of productions that gave us the original value of $S\phi$:

$$O \rightarrow u_1S_1v_1^{\text{rev}}, S_1 \rightarrow u_2S_2v_2^{\text{rev}}, \dots, S_k \rightarrow u_kS_kv_k^{\text{rev}},$$

which implies that $S\phi = (u_1u_2 \cdots u_k)^{-1}v_1v_2 \cdots v_k$, and the sequence that gave us $M\phi$:

$$O \rightarrow p_1M_1q_1^{\text{rev}}, M_1 \rightarrow p_2M_2q_2^{\text{rev}}, \dots, M_k \rightarrow p_lM_lq_l^{\text{rev}},$$

which implies that $M\phi = (p_1 p_2 \cdots p_l)^{-1} q_1 q_2 \cdots q_l$. Choose $r, s \in A^*$ such that $S \Rightarrow^* r \#_2 s^{\text{rev}}$. Then $L(\Gamma)$ contains both $u_1 \cdots u_k r \#_2 s^{\text{rev}} v_k^{\text{rev}} \cdots v_1^{\text{rev}}$ and (recalling that $M \rightarrow p S q^{\text{rev}}$ is a production) $p_1 \cdots p_l p r \#_2 s^{\text{rev}} q^{\text{rev}} q_l^{\text{rev}} \cdots q_1$. Suppose $u_1 \cdots u_k r = v_1 \cdots v_k s$ and $p_1 \cdots p_l p r = q_1 \cdots q_l q s$. Then $S\phi = (u_1 \cdots u_k)^{-1} v_1 \cdots v_k = r s^{-1} = (p_1 \cdots p_l p)^{-1} q_1 \cdots q_l q = p^{-1}(M\phi)q = z$, which is a contradiction.

Once we have iterated through all the productions of the form $M \rightarrow p S q^{\text{rev}}$, iterate through the productions of the form $M \rightarrow p \#_2 q^{\text{rev}}$, and check that $p^{-1}(M\phi)q$. If this check fails, halt: $L(\Gamma)$ does contain words $w \#_2 x^{\text{rev}}$ with $w \neq x$; the proof of this is very similar to the previous paragraph.

Finally, notice that if the iteration through all the productions completes with all the checks succeeding, a simple induction on derivations, using the values of $M\phi$, shows that all words $w \#_2 x^{\text{rev}} \in L(\Gamma)$ are such that $w = x$. 8.1

ALGORITHM 8.2.

Input: An interpreted word-hyperbolic structure Σ .

Output: If Σ describes a free semigroup, output *Yes*; otherwise output *No*.

Method:

1. For each $a \in A$, iterate the following:

- (a) Construct the context-free language

$$D_a = \{uv : u \#_1 v \#_2 a^{\text{rev}} \in M\}.$$

- (b) Check whether D_a is empty. If it is empty, proceed to the next iteration. If it is non-empty, choose some word $d_a \in D_a$. If d_a contains the letter a , halt and output *No*. If d_a does not contain the letter a , define the rational relations

$$\begin{aligned} \mathcal{Q}_L &= (\{(a, d_a)\} \cup \{(b, b) : b \in A - \{a\}\})^+ \\ \mathcal{Q}_M &= (\{(a, d_a)\} \cup \{(b, b) : b \in A - \{a\}\})^+ \#_1 \\ &\quad (\{(a, d_a)\} \cup \{(b, b) : b \in A - \{a\}\})^+ \#_2 \\ &\quad (\{(a, d_a^{\text{rev}})\} \cup \{(b, b) : b \in A - \{a\}\})^+. \end{aligned}$$

Modify Σ as follows: replace A by $A - \{a\}$; replace L by $L \circ \mathcal{Q}_L$; and replace M by $M \circ \mathcal{Q}_M$, and proceed to the next iteration.

2. If $L \neq A^+$, halt and output *No*.
3. Define the rational relation

$$\mathcal{P} = \{(a, a) : a \in A\} \cup \{(\#_1, \varepsilon), (\#_2, \#_2)\}.$$

Let $N = M \circ \mathcal{P}$. Using the method of [Lemma 8.1](#), check whether N contains any word of the form $x \#_2 w^{\text{rev}}$ with $x \neq w$. If so, halt and output *No*. Otherwise, halt and output *Yes*.

[Lemmata 8.3 to 8.5](#) show that this algorithm works.

LEMMA 8.3. *If Σ is a word-hyperbolic structure for a semigroup S , then the replacement Σ produced in step 1(b) is also a word-hyperbolic structure for a semigroup S .*

Proof of 8.3. If the language D_a is non-empty, then any word $w \in D_a$ is such that $\overline{w} = \overline{a}$. In particular, $\overline{d_a} = \overline{a}$. Furthermore, since $d_a \in (A - \{a\})^*$, we see that \overline{a} is a redundant generator. The rational relation Q_L relates any word in A^+ to the corresponding word in $(A - \{a\})^+$ with all instances of the symbol a replaced by the word d_a . The rational relation Q_M relates any word in $A^+ \#_1 A^+ \#_2 A^+$ to the corresponding word in $(A - \{a\})^+ \#_1 (A - \{a\})^+ \#_2 (A - \{a\})^+$ with all instances of the symbol a *before* $\#_2$ replaced by the word d_a and all instances of the symbol a *after* $\#_2$ replaced by the word d_a^{rev} . Hence

$$M \circ Q_M \subseteq (L \circ Q_L) \#_1 (L \circ Q_L) \#_2 (L \circ Q_L)^{\text{rev}}.$$

Since application of rational relations preserves regularity and context-freeness, $L \circ Q_L$ is regular and $M \circ Q_M$ is context-free. Finally, since $\overline{a} = \overline{d_a}$, we see that $L \circ Q_L$ maps onto S , and similarly $M \circ Q_M$ describes the multiplication of elements of S in terms of representatives in L . 8.3

LEMMA 8.4. *If Algorithm 8.2 outputs Yes, the semigroup defined by the word-hyperbolic structure Σ is a free semigroup.*

Proof of 8.4. The algorithm can only halt with output *Yes* in step 3, so the algorithm must pass step 2 as well. Hence the language of representatives is A^+ . Let S be the semigroup defined by L and let $\phi : A^+ \rightarrow S$ be an interpretation.

Suppose for *reductio ad absurdum* that ϕ is not injective. Then there are distinct words $u, v \in A^*$ such that $u\phi = v\phi$. Since $\phi|_A$ is injective by definition, at least one of u and v has length 2 or more. Interchanging u and v if necessary, assume $|u| \geq 2$. So $u = u'u''$, where u' and u'' are both non-empty. Since $L = A^+$, we have $u', u'' \in L$ and so $u' \#_1 u'' \#_2 v^{\text{rev}} \in M$. Hence $u \#_2 v^{\text{rev}} \in N$. But since the algorithm outputs *Yes* at step 3, there is no word $x \#_2 w^{\text{rev}} \in M$ with $x \neq w$. This is a contradiction and so ϕ is injective.

So $\phi : A^+ \rightarrow S$ is an isomorphism and so S is free. 8.4

LEMMA 8.5. *If the word-hyperbolic structure Σ defines a free semigroup, Algorithm 8.2 outputs Yes.*

Proof of 8.5. Let B^+ be the semigroup defined by Σ . Let $\phi : A^+ \rightarrow B^+$ be an interpretation. Since elements of B are indecomposable, $B \subseteq A\phi$.

In step 1, the algorithm iterates through each $a \in A$. For each $a \in B\phi^{-1} \subseteq A$, since $a\phi$ is indecomposable, the language D_a is empty and the algorithm moves to the next iteration.

Let $a \in A - B\phi^{-1}$. Then $a\phi$ has length (in B^+) at least two and so is decomposable. Hence there exist $u, v \in L$ such that $uv \in D_a$. Furthermore, since $u\phi$ and $v\phi$ must be shorter (in B^+) than $a\phi$, neither u nor v can include the letter a . Hence the replacement of Σ described in step 1(b) takes place. Since this occurs for all $a \in A - B\phi^{-1}$, at the end of step 1 we have a word-hyperbolic structure Σ with $A = B\phi^{-1}$. Since $\phi|_A$ is injective, $\phi|_A$ must be a bijection from A to B . Hence the homomorphism $\phi : A^+ \rightarrow B^+$ must be an isomorphism, and so $L = A^+$; thus the check in step 2 is successful. Therefore

$$M = \{u \#_1 v \#_2 (uv)^{\text{rev}} : u, v \in A^+\}$$

and so

$$M \circ \mathcal{P} = \{w \#_2 w^{\text{rev}} : w \in A^+\}.$$

Thus the check in step 3 is successful and the algorithm terminates with output *Yes*. 8.5

Thus, from [Lemmata 8.4](#) and [8.5](#), we obtain the decidability of freedom for word-hyperbolic semigroups:

THEOREM 8.6. *There is an algorithm that takes as input an interpreted word-hyperbolic structure Σ for a semigroup and decides whether it is a free semigroup.*

9 OPEN PROBLEMS

This concluding section lists some important question regarding decision problems for word-hyperbolic semigroups.

QUESTION 9.1. Is there an algorithm that takes as input an interpreted word-hyperbolic structure for a semigroup and decides whether that semigroup is (a) regular, (b) inverse?

Whether these properties are decidable for automatic semigroups is currently unknown.

QUESTION 9.2. Is there an algorithm that takes as input an interpreted word-hyperbolic structure for a semigroup and decides whether that semigroup is left-/right-/two-sided-cancellative?

Cancellativity and left-cancellativity are undecidable for automatic semigroups [[Caio6](#)]. Right-cancellativity is, however, decidable [[KO06](#), Corollary 3.3].

QUESTION 9.3. Is there an algorithm that takes as input an interpreted word-hyperbolic structure for a semigroup and decides whether that semigroup is finite?

The equivalent question for automatic semigroups is easy: one takes an automatic structure, effectively computes an automatic structure with uniqueness, and checks whether its regular language of representatives is finite. However, this approach cannot be used for word-hyperbolic semigroups, because there exist word-hyperbolic semigroups that do not admit word-hyperbolic structures with uniqueness indeed, they may not even admit regular languages of unique normal forms [[CM12](#), Examples 10 & 11].

QUESTION 9.4. Is there an algorithm that takes as input an interpreted word-hyperbolic structure for a semigroup and decides whether that semigroup admits a word-hyperbolic structure with uniqueness? (That is, where the language of representatives maps bijectively onto the semigroup.) If so, it is possible to compute a word-hyperbolic structure with uniqueness in this case?

10 REFERENCES

- [Caio6] A. J. Cain. ‘Cancellativity is undecidable for automatic semigroups’. *Q. J. Math.*, 57, no. 3 (2006), pp. 285–295. DOI: [10.1093/qmath/haio23](https://doi.org/10.1093/qmath/haio23).
- [CM12] A. J. Cain & V. Maltcev. ‘Context-free rewriting systems and word-hyperbolic structures with uniqueness’. *Internat. J. Algebra Comput.*, 22, no. 7 (2012). DOI: [10.1142/S0218196712500610](https://doi.org/10.1142/S0218196712500610).
- [CRRTo1] C. M. Campbell, E. F. Robertson, N. Ruškuc, & R. M. Thomas. ‘Automatic semigroups’. *Theoret. Comput. Sci.*, 250, no. 1–2 (2001), pp. 365–391. DOI: [10.1016/S0304-3975\(99\)00151-6](https://doi.org/10.1016/S0304-3975(99)00151-6).
- [DGo4] A. Duncan & R. H. Gilman. ‘Word hyperbolic semigroups’. *Math. Proc. Cambridge Philos. Soc.*, 136, no. 3 (2004), pp. 513–524. DOI: [10.1017/S0305004103007497](https://doi.org/10.1017/S0305004103007497).

- [ECH⁺92] D. B. A. Epstein, J. W. Cannon, D. F. Holt, S. V. F. Levy, M. S. Paterson, & W. P. Thurston. *Word Processing in Groups*. Jones & Bartlett, Boston, Mass., 1992.
- [FK04] J. Fountain & M. Kambites. ‘Hyperbolic groups and completely simple semigroups’. In *Semigroups and languages*, pp. 106–132. World Sci. Publ., River Edge, NJ, 2004. DOI: [10.1142/9789812702616_0007](https://doi.org/10.1142/9789812702616_0007).
- [Gil02] R. H. Gilman. ‘On the definition of word hyperbolic groups’. *Math. Z.*, 242, no. 3 (2002), pp. 529–541. DOI: [10.1007/s002090100356](https://doi.org/10.1007/s002090100356).
- [Gro87] M. Gromov. ‘Hyperbolic groups’. In S. M. Gersten, ed., *Essays in group theory*, vol. 8 of *Math. Sci. Res. Inst. Publ.*, pp. 75–263. Springer, New York, 1987.
- [HKOT02] M. Hoffmann, D. Kuske, F. Otto, & R. M. Thomas. ‘Some relatives of automatic and hyperbolic groups’. In G. M. S. Gomes, J. É. Pin, & P. V. Silva, eds, *Semigroups, Algorithms, Automata and Languages (Coimbra, 2001)*, pp. 379–406. World Scientific Publishing, River Edge, N.J., 2002.
- [How95] J. M. Howie. *Fundamentals of Semigroup Theory*, vol. 12 of *London Mathematical Society Monographs (New Series)*. Clarendon Press, Oxford University Press, New York, 1995.
- [HT03] M. Hoffmann & R. M. Thomas. ‘Notions of automaticity in semigroups’. *Semigroup Forum*, 66, no. 3 (2003), pp. 337–367. DOI: [10.1007/s002330010161](https://doi.org/10.1007/s002330010161).
- [HU79] J. E. Hopcroft & J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison–Wesley Publishing Co., Reading, Mass., 1979.
- [KO06] M. Kambites & F. Otto. ‘Uniform decision problems for automatic semigroups’. *J. Algebra*, 303, no. 2 (2006), pp. 789–809. DOI: [10.1016/j.jalgebra.2005.11.028](https://doi.org/10.1016/j.jalgebra.2005.11.028).
- [Otto7] F. Otto. ‘Some decidability and undecidability results on Green’s relations for automatic monoids’. *Semigroup Forum*, 75, no. 3 (2007), pp. 521–536.